

## NAME

SimilarityMatrixSDFFiles.pl - Calculate similarity matrices using fingerprints strings data in SDFFile(s)

## SYNOPSIS

SimilarityMatrixSDFFiles.pl SDFFile(s)...

```
SimilarityMatrixSDFFiles.pl [--alpha number] [--beta number] [-b, --BitVectorComparisonMode All |
"TanimotoSimilarity,[ TverskySimilarity, ... ]"] [--CompoundID DataFieldName or LabelPrefixString] [--CompoundIDMode
DataField | MolName | LabelPrefix | MolNameOrLabelPrefix] [-d, --detail InfoLevel] [-f, --fast] [--FingerprintsField
FieldLabel][-h, --help] [-m, --mode AutoDetect | FingerprintsBitVectorString | FingerprintsVectorString] [--OutDelim
comma | tab | semicolon] [--OutMatrixFormat RowsAndColumns | IDPairsAndValue] [-o, --overwrite] [-p, --precision
number] [-q, --quote Yes | No] [-r, --root RootName] [-v, --VectorComparisonMode All | "TanimotoSimilarity, [
ManhattanDistance, ...]" ] [--VectorComparisonFormulism All | AlgebraicForm | BinaryForm | SetTheoreticForm] [-w,
--WorkingDir dirname] SDFFile(s)...
```

## DESCRIPTION

Calculate similarity matrices using using fingerprint bit-vector or vector strings data field in *SDFFile(s)* and generate CSV/TSV text files containing values for specified similarity and distance coefficients.

Multiple SDFFile names are separated by spaces. The valid file extensions are *.sdf* and *.sd*. All other file names are ignored. All the SD files in a current directory can be specified either by *\*.sdf* or the current directory name.

## OPTIONS

--alpha *number*

Value of alpha parameter for calculating *Tversky* similarity coefficient specified for -b, --BitVectorComparisonMode option. It corresponds to weights assigned for bits set to "1" in a pair of fingerprint bit-vectors during the calculation of similarity coefficient. Possible values: *0 to 1*. Default value: <0.5>.

--beta *number*

Value of beta parameter for calculating *WeightedTanimoto* and *WeightedTversky* similarity coefficients specified for -b, --BitVectorComparisonMode option. It is used to weight the contributions of bits set to "0" during the calculation of similarity coefficients. Possible values: *0 to 1*. Default value of <1> makes *WeightedTanimoto* and *WeightedTversky* equivalent to *Tanimoto* and *Tversky*.

-b, --BitVectorComparisonMode *All* | "*TanimotoSimilarity*,[*TverskySimilarity*,...]"

Specify what similarity coefficients to use for calculating similarity matrices for fingerprints bit-vector strings data values in *TextFile(s)*: calculate similarity matrices for all supported similarity coefficients or specify a comma delimited list of similarity coefficients. Possible values: *All* | "*TanimotoSimilarity*,[*TverskySimilarity*,...]. Default: *TanimotoSimilarity*

*All* uses complete list of supported similarity coefficients: *BaroniUrbaniSimilarity*, *BuserSimilarity*, *CosineSimilarity*, *DiceSimilarity*, *DennisSimilarity*, *ForbesSimilarity*, *FossumSimilarity*, *HamannSimilarity*, *JacardSimilarity*, *Kulczynski1Similarity*, *Kulczynski2Similarity*, *MatchingSimilarity*, *McConnaugheySimilarity*, *OchiaiSimilarity*, *PearsonSimilarity*, *RogersTanimotoSimilarity*, *RussellRaoSimilarity*, *SimpsonSimilarity*, *SkoalSneath1Similarity*, *SkoalSneath2Similarity*, *SkoalSneath3Similarity*, *TanimotoSimilarity*, *TverskySimilarity*, *YuleSimilarity*, *WeightedTanimotoSimilarity*, *WeightedTverskySimilarity*. These similarity coefficients are described below.

For two fingerprint bit-vectors A and B of same size, let:

```
Na = Number of bits set to "1" in A
Nb = Number of bits set to "1" in B
Nc = Number of bits set to "1" in both A and B
Nd = Number of bits set to "0" in both A and B

Nt = Number of bits set to "1" or "0" in A or B (Size of A or B)
Nt = Na + Nb - Nc + Nd

Na - Nc = Number of bits set to "1" in A but not in B
Nb - Nc = Number of bits set to "1" in B but not in A
```

Then, various similarity coefficients [ Ref. 40 - 42 ] for a pair of bit-vectors A and B are defined as follows:

*BaroniUrbaniSimilarity*:  $(\text{SQRT}(Nc * Nd) + Nc) / (\text{SQRT}(Nc * Nd) + Nc + (Na - Nc) + (Nb - Nc))$  (

same as Buser )

*BuserSimilarity*:  $(\text{SQRT}(\text{Nc} * \text{Nd}) + \text{Nc}) / (\text{SQRT}(\text{Nc} * \text{Nd}) + \text{Nc} + (\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc}))$  ( same as BaroniUrbani )

*CosineSimilarity*:  $\text{Nc} / \text{SQRT}(\text{Na} * \text{Nb})$  (same as Ochiai)

*DiceSimilarity*:  $(2 * \text{Nc}) / (\text{Na} + \text{Nb})$

*DennisSimilarity*:  $(\text{Nc} * \text{Nd} - ((\text{Na} - \text{Nc}) * (\text{Nb} - \text{Nc}))) / \text{SQRT}(\text{Nt} * \text{Na} * \text{Nb})$

*ForbesSimilarity*:  $(\text{Nt} * \text{Nc}) / (\text{Na} * \text{Nb})$

*FossumSimilarity*:  $(\text{Nt} * ((\text{Nc} - 1/2) ** 2)) / (\text{Na} * \text{Nb})$

*HamannSimilarity*:  $((\text{Nc} + \text{Nd}) - (\text{Na} - \text{Nc}) - (\text{Nb} - \text{Nc})) / \text{Nt}$

*JaccardSimilarity*:  $\text{Nc} / ((\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc}) + \text{Nc}) = \text{Nc} / (\text{Na} + \text{Nb} - \text{Nc})$  (same as Tanimoto)

*Kulczynski1Similarity*:  $\text{Nc} / ((\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc})) = \text{Nc} / (\text{Na} + \text{Nb} - 2\text{Nc})$

*Kulczynski2Similarity*:  $((\text{Nc} / 2) * (2 * \text{Nc} + (\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc}))) / ((\text{Nc} + (\text{Na} - \text{Nc})) * (\text{Nc} + (\text{Nb} - \text{Nc}))) = 0.5 * (\text{Nc} / \text{Na} + \text{Nc} / \text{Nb})$

*MatchingSimilarity*:  $(\text{Nc} + \text{Nd}) / \text{Nt}$

*McConnaugheySimilarity*:  $(\text{Nc} ** 2 - (\text{Na} - \text{Nc}) * (\text{Nb} - \text{Nc})) / (\text{Na} * \text{Nb})$

*OchiaiSimilarity*:  $\text{Nc} / \text{SQRT}(\text{Na} * \text{Nb})$  (same as Cosine)

*PearsonSimilarity*:  $((\text{Nc} * \text{Nd}) - ((\text{Na} - \text{Nc}) * (\text{Nb} - \text{Nc}))) / \text{SQRT}(\text{Na} * \text{Nb} * (\text{Na} - \text{Nc} + \text{Nd}) * (\text{Nb} - \text{Nc} + \text{Nd}))$

*RogersTanimotoSimilarity*:  $(\text{Nc} + \text{Nd}) / ((\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc}) + \text{Nt}) = (\text{Nc} + \text{Nd}) / (\text{Na} + \text{Nb} - 2\text{Nc} + \text{Nt})$

*RussellRaoSimilarity*:  $\text{Nc} / \text{Nt}$

*SimpsonSimilarity*:  $\text{Nc} / \text{MIN}(\text{Na}, \text{Nb})$

*SkoalSneath1Similarity*:  $\text{Nc} / (\text{Nc} + 2 * (\text{Na} - \text{Nc}) + 2 * (\text{Nb} - \text{Nc})) = \text{Nc} / (2 * \text{Na} + 2 * \text{Nb} - 3 * \text{Nc})$

*SkoalSneath2Similarity*:  $(2 * \text{Nc} + 2 * \text{Nd}) / (\text{Nc} + \text{Nd} + \text{Nt})$

*SkoalSneath3Similarity*:  $(\text{Nc} + \text{Nd}) / ((\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc})) = (\text{Nc} + \text{Nd}) / (\text{Na} + \text{Nb} - 2 * \text{Nc})$

*TanimotoSimilarity*:  $\text{Nc} / ((\text{Na} - \text{Nc}) + (\text{Nb} - \text{Nc}) + \text{Nc}) = \text{Nc} / (\text{Na} + \text{Nb} - \text{Nc})$  (same as Jaccard)

*TverskySimilarity*:  $\text{Nc} / (\text{alpha} * (\text{Na} - \text{Nc}) + (1 - \text{alpha}) * (\text{Nb} - \text{Nc}) + \text{Nc}) = \text{Nc} / (\text{alpha} * (\text{Na} - \text{Nb}) + \text{Nb})$

*YuleSimilarity*:  $((\text{Nc} * \text{Nd}) - ((\text{Na} - \text{Nc}) * (\text{Nb} - \text{Nc}))) / ((\text{Nc} * \text{Nd}) + ((\text{Na} - \text{Nc}) * (\text{Nb} - \text{Nc})))$

Values of Tanimoto/Jaccard and Tversky coefficients are dependent on only those bit which are set to "1" in both A and B. In order to take into account all bit positions, modified versions of Tanimoto [ Ref. 42 ] and Tversky [ Ref. 43 ] have been developed.

Let:

$\text{Na}'$  = Number of bits set to "0" in A

$\text{Nb}'$  = Number of bits set to "0" in B

$\text{Nc}'$  = Number of bits set to "0" in both A and B

Tanimoto':  $\text{Nc}' / ((\text{Na}' - \text{Nc}') + (\text{Nb}' - \text{Nc}') + \text{Nc}') = \text{Nc}' / (\text{Na}' + \text{Nb}' - \text{Nc}')$

Tversky':  $\text{Nc}' / (\text{alpha} * (\text{Na}' - \text{Nc}') + (1 - \text{alpha}) * (\text{Nb}' - \text{Nc}') + \text{Nc}') = \text{Nc}' / (\text{alpha} * (\text{Na}' - \text{Nb}') + \text{Nb}')$

Then:

*WeightedTanimotoSimilarity* =  $\text{beta} * \text{Tanimoto} + (1 - \text{beta}) * \text{Tanimoto}'$

*WeightedTverskySimilarity* =  $\text{beta} * \text{Tversky} + (1 - \text{beta}) * \text{Tversky}'$

#### --CompoundID *DataFieldName* or *LabelPrefixString*

This value is --CompoundIDMode specific and indicates how compound ID is generated.

For *DataField* value of --CompoundIDMode option, this option corresponds to datafield label name whose value is used as compound ID; otherwise, it's a prefix string used for generating compound IDs like *LabelPrefixString*<Number>. Default value, *Cmpd*, generates compound IDs which look like *Cmpd*<Number>.

Examples for *DataField* value of --CompoundIDMode:







Yes.

**-r, --root *RootName***

New file name is generated using the root: <Root><BitVectorComparisonMode>.<Ext> or <Root><VectorComparisonMode><VectorComparisonFormulism>.<Ext>. The csv, and tsv <Ext> values are used for comma/semicolon, and tab delimited text files respectively. This option is ignored for multiple input files.

**-v, --VectorComparisonMode *All | "TanimotoSimilarity,[ManhattanDistance,...]"***

Specify what similarity or distance coefficients to use for calculating similarity matrices for fingerprint vector strings data values in *TextFile(s)*: calculate similarity matrices for all supported similarity and distance coefficients or specify a comma delimited list of similarity and distance coefficients. Possible values: *All | "TanimotoSimilarity,[ManhattanDistance,...]"*. Default: *TanimotoSimilarity*.

The value of *-v, --VectorComparisonMode*, in conjunction with *--VectorComparisonFormulism*, decides which type of similarity and distance coefficient formulism gets used.

*All* uses complete list of supported similarity and distance coefficients: *CosineSimilarity, CzekanowskiSimilarity, DiceSimilarity, OchiaiSimilarity, JaccardSimilarity, SorensonSimilarity, TanimotoSimilarity, CityBlockDistance, EuclideanDistance, HammingDistance, ManhattanDistance, SoergelDistance*. These similarity and distance coefficients are described below.

FingerprintsVector.pm module, used to calculate similarity and distance coefficients, provides support to perform comparison between vectors containing three different types of values:

Type I: OrderedNumericalValues

- . Size of two vectors are same
- . Vectors contain real values in a specific order. For example: MACCS keys count, Topological pharmacophore atom pairs and so on.

Type II: UnorderedNumericalValues

- . Size of two vectors might not be same
- . Vectors contain unordered real value identified by value IDs. For example: Topological atom pairs, Topological atom torsions and so on

Type III: AlphaNumericalValues

- . Size of two vectors might not be same
- . Vectors contain unordered alphanumerical values. For example: Extended connectivity fingerprints, atom neighborhood fingerprints.

Before performing similarity or distance calculations between vectors containing UnorderedNumericalValues or AlphaNumericalValues, the vectors are transformed into vectors containing unique OrderedNumericalValues using value IDs for UnorderedNumericalValues and values itself for AlphaNumericalValues.

Three forms of similarity and distance calculation between two vectors, specified using *--VectorComparisonFormulism* option, are supported: *AlgebraicForm, BinaryForm or SetTheoreticForm*.

For *BinaryForm*, the ordered list of processed final vector values containing the value or count of each unique value type is simply converted into a binary vector containing 1s and 0s corresponding to presence or absence of values before calculating similarity or distance between two vectors.

For two fingerprint vectors A and B of same size containing OrderedNumericalValues, let:

$N = \text{Number values in A or B}$

$X_a = \text{Values of vector A}$

$X_b = \text{Values of vector B}$

$X_{ai} = \text{Value of } i\text{th element in A}$

$X_{bi} = \text{Value of } i\text{th element in B}$

$\text{SUM} = \text{Sum of } i \text{ over } N \text{ values}$

For SetTheoreticForm of calculation between two vectors, let:

$\text{SetIntersection}X_aX_b = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) )$

$$\text{SetDifferenceXaXb} = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) )$$

For BinaryForm of calculation between two vectors, let:

$$\begin{aligned} N_a &= \text{Number of bits set to "1" in A} = \text{SUM} ( X_{ai} ) \\ N_b &= \text{Number of bits set to "1" in B} = \text{SUM} ( X_{bi} ) \\ N_c &= \text{Number of bits set to "1" in both A and B} = \text{SUM} ( X_{ai} * X_{bi} ) \\ N_d &= \text{Number of bits set to "0" in both A and B} \\ &= \text{SUM} ( 1 - X_{ai} - X_{bi} + X_{ai} * X_{bi} ) \end{aligned}$$

$$N = \text{Number of bits set to "1" or "0" in A or B} = \text{Size of A or B} = N_a + N_b - N_c + N_d$$

Additionally, for BinaryForm various values also correspond to:

$$\begin{aligned} N_a &= | X_a | \\ N_b &= | X_b | \\ N_c &= | \text{SetIntersectionXaXb} | \\ N_d &= N - | \text{SetDifferenceXaXb} | \end{aligned}$$

$$\begin{aligned} | \text{SetDifferenceXaXb} | &= N - N_d = N_a + N_b - N_c + N_d - N_d = N_a + N_b - N_c \\ &= | X_a | + | X_b | - | \text{SetIntersectionXaXb} | \end{aligned}$$

Various distance and similarity coefficients [ Ref 40, Ref 62, Ref 64 ] for a pair of vectors A and B in *AlgebraicForm*, *BinaryForm* and *SetTheoreticForm* are defined as follows:

CityBlockDistance: ( same as HammingDistance and ManhattanDistance)

$$\text{AlgebraicForm: } \text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) )$$

$$\text{BinaryForm: } ( N_a - N_c ) + ( N_b - N_c ) = N_a + N_b - 2 * N_c$$

$$\text{SetTheoreticForm: } | \text{SetDifferenceXaXb} | - | \text{SetIntersectionXaXb} | = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$$

CosineSimilarity: ( same as OchiaiSimilarityCoefficient)

$$\text{AlgebraicForm: } \text{SUM} ( X_{ai} * X_{bi} ) / \text{SQRT} ( \text{SUM} ( X_{ai} ** 2 ) * \text{SUM} ( X_{bi} ** 2 ) )$$

$$\text{BinaryForm: } N_c / \text{SQRT} ( N_a * N_b )$$

$$\text{SetTheoreticForm: } | \text{SetIntersectionXaXb} | / \text{SQRT} ( |X_a| * |X_b| ) = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / \text{SQRT} ( \text{SUM} ( X_{ai} ) * \text{SUM} ( X_{bi} ) )$$

CzekanowskiSimilarity: ( same as DiceSimilarity and SorensonSimilarity)

$$\text{AlgebraicForm: } ( 2 * ( \text{SUM} ( X_{ai} * X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) )$$

$$\text{BinaryForm: } 2 * N_c / ( N_a + N_b )$$

$$\text{SetTheoreticForm: } 2 * | \text{SetIntersectionXaXb} | / ( |X_a| + |X_b| ) = 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) )$$

DiceSimilarity: ( same as CzekanowskiSimilarity and SorensonSimilarity)

$$\text{AlgebraicForm: } ( 2 * ( \text{SUM} ( X_{ai} * X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) )$$

$$\text{BinaryForm: } 2 * N_c / ( N_a + N_b )$$

$$\text{SetTheoreticForm: } 2 * | \text{SetIntersectionXaXb} | / ( |X_a| + |X_b| ) = 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) )$$

EuclideanDistance:

$$\text{AlgebraicForm: } \text{SQRT} ( \text{SUM} ( ( ( X_{ai} - X_{bi} ) ** 2 ) ) )$$

$$\text{BinaryForm: } \text{SQRT} ( ( N_a - N_c ) + ( N_b - N_c ) ) = \text{SQRT} ( N_a + N_b - 2 * N_c )$$

$$\text{SetTheoreticForm: } \text{SQRT} ( | \text{SetDifferenceXaXb} | - | \text{SetIntersectionXaXb} | ) = \text{SQRT} ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) )$$

HammingDistance: ( same as CityBlockDistance and ManhattanDistance)

$$\text{AlgebraicForm: } \text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) )$$

$$\text{BinaryForm: } ( N_a - N_c ) + ( N_b - N_c ) = N_a + N_b - 2 * N_c$$

$$\text{SetTheoreticForm: } | \text{SetDifferenceXaXb} | - | \text{SetIntersectionXaXb} | = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$$

JaccardSimilarity: ( same as TanimotoSimilarity)

*AlgebraicForm*:  $\text{SUM} ( X_{ai} * X_{bi} ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) - \text{SUM} ( X_{ai} * X_{bi} ) )$

*BinaryForm*:  $N_c / ( ( N_a - N_c ) + ( N_b - N_c ) + N_c ) = N_c / ( N_a + N_b - N_c )$

*SetTheoreticForm*:  $| \text{SetIntersectionXaXb} | / | \text{SetDifferenceXaXb} | = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

ManhattanDistance: ( same as CityBlockDistance and HammingDistance)

*AlgebraicForm*:  $\text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) )$

*BinaryForm*:  $( N_a - N_c ) + ( N_b - N_c ) = N_a + N_b - 2 * N_c$

*SetTheoreticForm*:  $| \text{SetDifferenceXaXb} | - | \text{SetIntersectionXaXb} | = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

OchiaiSimilarity: ( same as CosineSimilarity)

*AlgebraicForm*:  $\text{SUM} ( X_{ai} * X_{bi} ) / \text{SQRT} ( \text{SUM} ( X_{ai} ** 2 ) * \text{SUM} ( X_{bi} ** 2 ) )$

*BinaryForm*:  $N_c / \text{SQRT} ( N_a * N_b )$

*SetTheoreticForm*:  $| \text{SetIntersectionXaXb} | / \text{SQRT} ( |X_a| * |X_b| ) = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / \text{SQRT} ( \text{SUM} ( X_{ai} ) * \text{SUM} ( X_{bi} ) )$

SorensonSimilarity: ( same as CzekanowskiSimilarity and DiceSimilarity)

*AlgebraicForm*:  $( 2 * ( \text{SUM} ( X_{ai} * X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) )$

*BinaryForm*:  $2 * N_c / ( N_a + N_b )$

*SetTheoreticForm*:  $2 * | \text{SetIntersectionXaXb} | / ( |X_a| + |X_b| ) = 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) )$

SoergelDistance:

*AlgebraicForm*:  $\text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) ) / \text{SUM} ( \text{MAX} ( X_{ai}, X_{bi} ) )$

*BinaryForm*:  $1 - N_c / ( N_a + N_b - N_c ) = ( N_a + N_b - 2 * N_c ) / ( N_a + N_b - N_c )$

*SetTheoreticForm*:  $( | \text{SetDifferenceXaXb} | - | \text{SetIntersectionXaXb} | ) / | \text{SetDifferenceXaXb} | = ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

TanimotoSimilarity: ( same as JaccardSimilarity)

*AlgebraicForm*:  $\text{SUM} ( X_{ai} * X_{bi} ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) - \text{SUM} ( X_{ai} * X_{bi} ) )$

*BinaryForm*:  $N_c / ( ( N_a - N_c ) + ( N_b - N_c ) + N_c ) = N_c / ( N_a + N_b - N_c )$

*SetTheoreticForm*:  $| \text{SetIntersectionXaXb} | / | \text{SetDifferenceXaXb} | = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

--VectorComparisonFormulism *All* | "*AlgebraicForm*,[*BinaryForm*,*SetTheoreticForm*]"

Specify fingerprints vector comparison formulism to use for calculation similarity and distance coefficients during -v, --VectorComparisonMode: use all supported comparison formulisms or specify a comma delimited. Possible values: *All* | "*AlgebraicForm*,[*BinaryForm*,*SetTheoreticForm*]". Default value: *AlgebraicForm*.

*All* uses all three forms of supported vector comparison formulism for values of -v, --VectorComparisonMode option.

For fingerprint vector strings containing AlphaNumericalValues data values - ExtendedConnectivityFingerprints, AtomNeighborhoodsFingerprints and so on - all three formulism result in same value during similarity and distance calculations.

-w, --WorkingDir *DirName*

Location of working directory. Default: current directory.

## EXAMPLES

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPHexTanimotoSimilarity.csv file containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -o SampleFPHex.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient using algebraic formulism for fingerprints vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPCountTanimotoSimilarityAlgebraicForm.csv file containing sequentially

generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -o SampleFPCount.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPHexTanimotoSimilarity.csv file in IDPairsAndValue format containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl --OutMatrixFormat IDPairsAndValue -o
SampleFPHex.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPHexTanimotoSimilarity.csv file containing compound IDs from mol name line, type:

```
% SimilarityMatrixSDFFiles.pl --CompoundIDMode MolName -o SampleFPHex.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPHexTanimotoSimilarity.csv file containing compound IDs from data field name Mol\_ID, type:

```
% SimilarityMatrixSDFFiles.pl --CompoundIDMode Data Field
--CompoundID Mol_ID -o SampleFPBin.sdf
```

To generate similarity matrices corresponding to Buser, Dice and Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create SampleFPBin[CoefficientName]Similarity.csv files containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -b "BuserSimilarity,DiceSimilarity,
TanimotoSimilarity" -o SampleFPBin.sdf
```

To generate similarity matrices corresponding to CityBlock distance Tanimoto similarity coefficients using algebraic formulism for fingerprints vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create SampleFPCount[CoefficientName]AlgebraicForm.csv files containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -v "CityBlockDistance,TanimotoSimilarity"
-o SampleFPCount.sdf
```

To generate similarity matrices corresponding to CityBlock distance Tanimoto similarity coefficients using binary formulism for fingerprints vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create SampleFPCount[CoefficientName]Binary.csv files containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -v "CityBlockDistance,TanimotoSimilarity"
--VectorComparisonFormulism BinaryForm -o SampleFPCount.sdf
```

To generate similarity matrices corresponding to CityBlock distance Tanimoto similarity coefficients using all supported comparison formulisms for fingerprints vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create SampleFPCount[CoefficientName][FormulismName].csv files containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -v "CityBlockDistance,TanimotoSimilarity"
--VectorComparisonFormulism All -o SampleFPCount.sdf
```

To generate similarity matrices corresponding to all available similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create SampleFPHex[CoefficientName].csv files containing sequentially generated compound IDs with Cmpd prefix, type

---

```
% SimilarityMatrixSDFFiles.pl -m AutoDetect --BitVectorComparisonMode All
--alpha 0.5 -beta 0.5 -o SampleFPHex.sdf
```

To generate similarity matrices corresponding to all available similarity and distance coefficients using all comparison formulism for fingerprints vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create SampleFPCount[CoefficientName][FormulismName].csv files containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -m AutoDetect --VectorComparisonMode All
--VectorComparisonFormulism All -o SampleFPCount.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field name Fingerprints and create a SampleFPHexTanimotoSimilarity.csv file containing compound IDs present in data field name Mol\_ID, type:

```
% SimilarityMatrixSDFFiles.pl --FingerprintsField Fingerprints
--CompoundIDMode Data Field --CompoundID Mol_ID -o SampleFPHex.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPHexTanimotoSimilarity.csv file containing compound IDs from molname line or sequentially generated compound IDs with Mol prefix, type:

```
% SimilarityMatrixSDFFiles.pl --CompoundIDMode MolnameOrLabelPrefix
--CompoundID Mol -o SampleFPHex.sdf
```

To generate a similarity matrix corresponding to Tanimoto similarity coefficient for fingerprints bit-vector strings data corresponding to supported fingerprints present in a data field with Fingerprint substring in its label and create a SampleFPHexTanimotoSimilarity.tsv file containing sequentially generated compound IDs with Cmpd prefix, type:

```
% SimilarityMatrixSDFFiles.pl -OutDelim Tab --quote No -o SampleFPHex.sdf
```

## AUTHOR

Manish Sud <msud@san.rr.com>

## SEE ALSO

InfoFingerprintsTextFiles.pl, InfoFingerprintsSDFFiles.pl, SimilarityMatrixTextFiles.pl, AtomNeighborhoodsFingerprints.pl, ExtendedConnectivityFingerprints.pl, MACCSKeysFingerprints.pl, PathLengthFingerprints.pl, TopologicalAtomPairsFingerprints.pl, TopologicalAtomTorsionsFingerprints.pl, TopologicalPharmacophoreAtomPairsFingerprints.pl, TopologicalPharmacophoreAtomTripletsFingerprints.pl

## COPYRIGHT

Copyright (C) 2004-2010 Manish Sud. All rights reserved.

This file is part of MayaChemTools.

MayaChemTools is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.